UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/830,042	04/23/2004	John M. Holt	22216-00003-US1	9699

22918 7590 12/31/2007

PERKINS COIE LLP
P.O. BOX 2168
MENLO PARK, CA 94026

EXAMINER	
RUTTEN, JAMES D	
ART UNIT	**PAPER NUMBER**
2192	
MAIL DATE	**DELIVERY MODE**
12/31/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/830,042	HOLT, JOHN M.	
	Examiner	Art Unit	
	J. Derek Rutten	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *21 November 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-9,11-18 and 20-28* is/are pending in the application.

4a) Of the above claim(s) *1-6,11-13 and 20-23* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *7-9, 14-18, and 24-28* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This action is in response to Applicant's submission filed 5/25/07, responding to the 1/25/07 Office action which detailed the rejection of claims 7-10 and 14-19. Claims 7, 8, and 14 have been amended, claims 10 and 19 have been canceled, and claims 1-6, 11-13, and 20-23 have been withdrawn. Claims 1-9, 11-18, and 20-23 remain pending in the application. Claims 7-9 and 14-18 have been fully considered and examined.

Response to Amendment/Arguments

2. Applicant's 11/21/07 amendment has obviate the objections to claims 7 and 14, the rejections to claims 7-9 and 14-18 under 35 U.S.C. § 112, second paragraph. Therefore, these objections and rejections are withdrawn.

3. Applicant's arguments, see pages 8-11, filed 11/21/07, with respect to the rejection(s) of claim(s) 7, 8, 14, and 15 under 35 U.S.C. 102 have been fully considered and are persuasive. Therefore, the rejections have been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of prior art of record "MultiJav: A Distributed Shared Memory System Based on Multiple Java Virtual Machines" by Chen et al.

4. Applicant's arguments, see pages 12 and 13, filed 11/21/07 have been fully considered but they are not persuasive.

In response to applicant's argument that the examiner's conclusion of obviousness is based upon improper hindsight reasoning, it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the

time the claimed invention was made, and does not include knowledge gleaned only from the applicant's disclosure, such a reconstruction is proper. See *In re McLaughlin*, 443 F.2d 1392, 170 USPQ 209 (CCPA 1971).

In response to applicant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988)and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, Dimpsey provides a teaching that compilation techniques may provide an increase in execution speed. Thus, Dimpsey provides the motivation to combine with Scales.

Claim Objections

5. Claims 14 and 28 are objected to because of the following informalities: Claim 14 appears to contain a typo in the word "interconnecatable" (line 3 of claim 14 and line 2 of claim 28) which should instead be --interconnectable--.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 7, 8, 24, and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent 6,760,903 to Morshed et al. (hereinafter "Morshed") in view of "MultiJav: A Distributed Shared Memory System Based on Multiple Java Virtual Machines" by Chen et al. (hereinafter "Chen").

In regard to claim 7, Morshed discloses:

A method of loading an application program ...onto each of a plurality of computers, See column 20:60-61:

> Byte code may be instrumented by instrumenting each class as the class is loaded by the VM runtime system.

the plurality of computers being interconnectable via a communications link, See Figure 29 and associated text in column 32:50-57, e.g. "network";

and different portions of said application program being simultaneously executable on each different one of the plurality of computers with each different one of the plurality of computers having a different independent local memory accessible only by a corresponding portion of the application program, See column 34 lines 39-42:

> In this example, the COM DLL may be used for facilitating interprocess communication, for example, as between a client and a server as well as between any two server systems.

Also see FIG. 29 and associated text in column 32 lines 50-57, e.g. elements 1000 and 1016a. This figure depicts separate computer systems which Morshed describes in terms of Intel Pentium processors running the MS Windows® operating systems which implicitly provides access to independent local memory by a corresponding portion of software.

the method comprising the steps of:

loading the application program onto each different computer of said plurality of computers; and See column 20:60-61, also column 34 lines 39-42 as cited above.

modifying the application program on each said different computer before execution of said corresponding portion of the application program on each said different computer. See column 20:60-61:

> Byte code may be instrumented by instrumenting each class as the class is loaded by the VM runtime system

Morshed does not expressly disclose: *written to operate only on a single computer*. However, Chen teaches loading an application written to operate only on a single computer, on different computers. See page 1 right column: "Thus, the same code can be run on a standalone machine without modification." It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Chen's teaching of simultaneous execution of uniprocessor programs with Morshed's distribution in order to provide distributed computing while maintaining portability and adherence to standard specifications as suggested by Chen (see page 1 right column).

In regard to claim 8, the above rejection of claim 7 is incorporated. Morshed further discloses: *wherein the step of modifying the application program is different for different computers.* See column 33:28-31, e.g. "different activities."

In regard to claim 24, the above rejection of claim 7 is incorporated. Morshed does not expressly disclose: *wherein said program written to operate on only a single*

computer is a program written to execute within a local processor or processors and local memory coupled to the processor or processors within the single computer. However, Chen teaches this on page 1 right column. Chen's "standalone machine" execution necessarily executes within a local processor using local memory.

In regard to claim 27, Morshed discloses:

said different independent local memory within each said different computer being accessible during execution of said application program and said different portions of said application program only by the different portion of the application program actually executing within the different computer, See Morshed column 34 lines 39-42:

> In this example, the COM DLL may be used for facilitating interprocess communication, for example, as between a client and a server as well as between any two server systems.

Note that clients and servers are distinct systems that only control their respective local resources. All further limitations have been addressed in the above rejection of claim 7.

8. Claim 9 is rejected under 35 U.S.C. 103(a) as being unpatentable over Morshed and Chen in view of U.S. Patent No. 5,802,585 to Scales et al. (hereinafter "Scales").

In regard to claim 9, the above rejection of claim 7 or 8 is incorporated. Morshed and Chen do not expressly disclose further elements of claim 9. However, Scales teaches: *wherein said modifying step comprises:*

(i) detecting instructions which share memory records See Scales column 4:38-42, e.g. "coherency."

(ii) listing all such shared memory records and providing a naming tag for each listed memory record See column 6:20-21, e.g. "table," and column 11:6-10, e.g. "ID."

(iii) detecting those instructions which write to, or manipulate the contexts of, any of said listed memory records, and See column 4:30-32, e.g. "stores."

(iv) generating an alert instruction corresponding to each said detected write or manipulate instruction, said alert instruction forwarding the re-written or manipulated contents and name tag of each said re-written or manipulated listed memory record. See column 1:43-49, e.g. "message passing" and column 32-35, e.g. "miss check."

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Scales' shared memory with Morshed's loading in order to provide coherency (see Scales column 1:20-26).

9. Claims 14, 15, 26, and 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Scales in view of Chen.

In regard to claim 14, Scales discloses:

A method of compiling or modifying an application program ...to run simultaneously on one of a plurality of computers interconnectable via a communications link See Figure 2 with supporting disclosure in column 4:29-30, e.g. "programs 215 are instrumented", Also see column 4 lines 18-19: "During operation of the system 200, instructions of the programs 215 are executed by the processors 211." This passage shows that a portion of the program runs on one of a plurality of computers..

with different portions of said application program being simultaneously executable on different ones of said plurality of computers with each one of the plurality of computers having an independent local memory accessible only by the corresponding portion of the application program, See column 5 lines 6-8, e.g. "store private data only operated on by a local processor."

said method comprising the steps of:

(i) detecting instructions which share memory records See Scales column 4:38-42, e.g. "coherency."

(ii) listing all such shared memory records and providing a naming tag for each listed memory record See column 6:20-21, e.g. "table," and column 11:6-10, e.g. "ID."

(iii) detecting those instructions which write to, or manipulate the contents of, any of said listed memory records, and See column 4:30-32, e.g. "stores."

(iv) generating an alert instruction following each said detected write or manipulate instruction, said alert instruction forwarding the re-written or manipulated contents and name tag of each said re-written or manipulated listed memory record. See column 1:43-49, e.g. "message passing" and column 32-35, e.g. "miss check."

Scales does not expressly disclose: *written to operate only on a single computer.* However, Chen teaches loading an application written to operate only on a single computer, on different computers. See page 1 right column: "Thus, the same code can be run on a standalone machine without modification." It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Chen's teaching of simultaneous execution of uniprocessor programs with Scales' distribution in order to

provide distributed computing while maintaining portability and adherence to standard specifications as suggested by Chen (see page 1 right column).

In regard to claim 15, the above rejection of claim 14 is incorporated. Scales further discloses: *carried out prior to loading the application program onto each said computer.* See column 4 lines 29-30, e.g. "prior to execution."

In regard to claim 26, the above rejection of claim 14 is incorporated. Scales does not expressly disclose: *wherein said program written to operate on only a single computer is a program written to execute within a local processor or processors and local memory coupled to the processor or processors within the single computer.* However, Chen teaches this on page 1 right column. Chen's "standalone machine" execution necessarily executes within a local processor using local memory.

In regard to claim 28, Scales does not expressly disclose: *said different independent local memory within each said different computer being accessible during execution of said application program and said different portions of said application program only by the different portion of the application program actually executing within the different computer,* However, Chen teaches this through a memory coherence model. See section 3.3 on page 5: "The memory of site *q* is required to be consistent with site *p*, which requires all the updates to shared variables at site *p* to be visible at site *q*." It would have been obvious to one of ordinary skill in the art at the time the invention

was made to use Chen's memory coherence with Scales distributed execution in order to provide data coherency as suggested by Chen. All further limitations have been addressed in the above rejection of claim 14.

10. Claim 16 is rejected under 35 U.S.C. 103(a) as being unpatentable over Scales and Chen as applied to claim 14 above, and further in view of Morshed.

In regard to claim 16, the above rejection of claim 14 is incorporated. Scales and Chen do not expressly disclose: *carried out during loading of the application program onto each said computer.* However, Morshed teaches instrumenting during loading. See column 20:60-61. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Morshed's teaching of loading with Scales' modification in order to automatically instrument a class instance (see Morshed column 20:61-65).

11. Claims 17 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Scales and Chen as applied to claim 14 above, and further in view of U.S. Patent Application Publication 2004/0163077 by Dimpsey et al. (hereinafter "Dimpsey.")

In regard to claim 17, the above rejection of claim 14 is incorporated. Scales and Chen do not expressly disclose: *carried out by just-in-time compilation.* However, Dimpsey teaches just-in-time compilation. See paragraph [0050]. It would have been

obvious to one of ordinary skill in the art at the time the invention was made to use Dimpsey's compiler with Scales' modification in order to increase execution speed while reducing compilation time as inherently provided by just-in-time compilation.

In regard to claim 18, the above rejection of claim 14 is incorporated. Scales and Chen do not expressly disclose: *carried out by re-compilation after loading.* However, Dimpsey teaches dynamic instrumentation after loading code. See Abstract. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Dimpsey's dynamic instrumentation in order to minimize system perturbation (see Dimpsey's Abstract). Note that page 9 lines 6-8 of Applicant's specification inform broad interpretation of the concept of "compilation."

12. Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Morshed and Chen as applied to claim 7 above, and further in view of U.S. Patent No. 6,862,608 to Buhlman et al. (hereinafter "Buhlman").

In regard to claim 25, the above rejection of claim 7 is incorporated. Morshed and Chen do not expressly disclose: *wherein each of the computers operates with the same application program and data and thus all of the plurality of computers have the same application program and data.* However, Buhlman teaches that programs can be operated using the same program and data. See column 1 lines 30-38. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use

Buhlman's teaching of multiple copies of shared memory with Morshed's distributed execution in order to reduce latency as suggested by Buhlman.

Conclusion

13. Any inquiry concerning this communication or earlier communications from the examiner should be directed to J. Derek Rutten whose telephone number is (571)272-3703. The examiner can normally be reached on M-F 8:00-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571)272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/J. Derek Rutten/
Patent Examiner, AU 2192